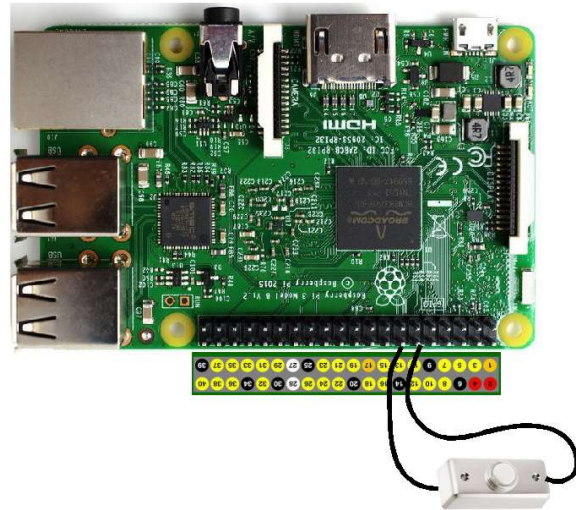
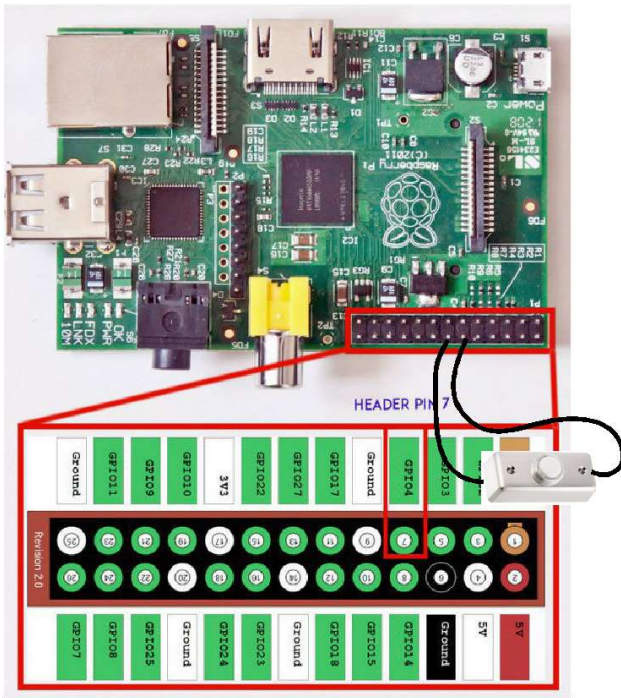


Koisk_flip.py

9/7/23

The program will flip between displaying Spanish and English .png files when either any mouse button is pressed or the “door bell” momentary push button switch is pressed.



¡Aprende en Casa!

Interés	Sugerencia
Programación en Computadora	Code.org, Codecademy.com, KhanAcademy.com: clases en línea Scratch, Small Basic, Kodu: free desktop downloads TouchDevelop: gratis en línea o descargable maker de aplicaciones celulares
Electronica	Adafruit.com, Sparkfun.com: piezas, kits, tutorios gratis (algunos en español) apuntesdeelectronica.com : tutorios gratis en español
Robotica	Instructables.com, LetsMakeRobots.com: tutorias gratis
Modelado y Impresión 3D	Programas gratis de modelado: Sculptiris, TinkerCad, Blender Impresoras baratas que nos gustan: \$180 Monoprice MP Mini Delta 3D Printer (bit.ly/vislalab180) o la más avanzada \$350 Prusa Mini Plus (tinyurl.com/vislalab-prusa)
Desarrollo de Videojuegos 3D	Unity3D, Unreal Engine: descargas gratuitas

¡Aprende de cero a programar y hacer tu primer robot simple con nuestras clases!
<https://tinyurl.com/vislalab-arduino>

Nuestros cursos gratis en línea (en Inglés)

Siganos en TWITTER!
twitter.com/MathewL.Faerber
#vialabNRC

Learn @ Home!

Interest	Suggestions
Computer Programming	Code.org, Codecademy.com, KhanAcademy.com: free online classes Scratch, Small Basic, Kodu: free desktop downloads TouchDevelop: free online or downloadable phone app maker
Electronics	Adafruit.com, Sparkfun.com: parts, kits, free tutorials
Robotics	Instructables.com, LetsMakeRobots.com: free tutorials
3D Modeling & Printing	Free 3D Modeling Programs: Sculptiris, TinkerCad, Blender Cheapest printers we recommend: \$180 Monoprice MP Mini Delta 3D Printer (bit.ly/vislalab180) or the advanced \$350 Prusa Mini Plus (tinyurl.com/vislalab-prusa)
3D Game Development	Unity3D, Unreal Engine: free downloads

Learn how to do coding for the first time and build a simple robot!
<https://tinyurl.com/vislalab-arduino>

FOLLOW US ON TWITTER!
twitter.com/MathewL.Faerber
#vialabNRC

Koisk_flip.py

It runs on a Raspberry Pi and uses pygame. I have run it in Windows after pygame has been installed.

Files needed in the */home/pi/kiosk* folder:

kiosk_flip.py

ResouceEnglish.png

ResourceSpanish.png

The artwork of the two Resource files is 960x720

On the Raspberry Pi, set the display to 1280x1024

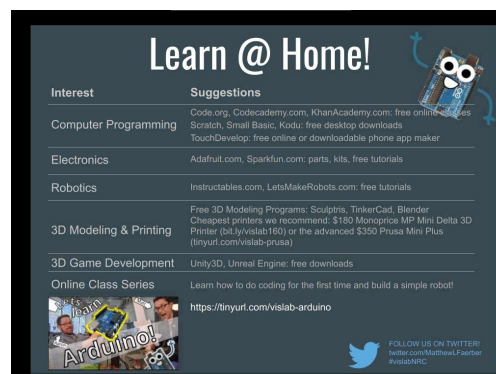
sudo raspi-config

7 Advanced Options

A5 Resolution

DMT Mode 35 1280x1024 60Hz 5:4

If the resolution is changed to **Mode 16 1024 x 768**, the display is closer to full screen:



Koisk_flip.py

The switch is connected to pins 12 and 14 on the Rasp Pi as shown above.

To run the program, switch to the */home/pi/kiosk* folder.

Enter: ***sudo python kiosk_flipx26.py.***

Note: we need to use ***sudo*** because the ***RPi.GPIO*** library was imported.

When running the FULLSCREEN version, you can do nothing else. To get out of the FULLSCREEN mode, you have to reboot the Raspberry Pi.

If the Raspberry Pi is in autostart mode, you have to follow the procedure below.

Here is the procedure:

Reboot the unit by powering down and then re-starting.

While the ***autoexec.sh*** file is launching the python script, stop the execution using **Ctrl-c**.

To set up the program to run when the Raspberry Pi is powered up. Follow the procedure in the link below:

https://dunn-itvideoservicesllc.yolasite.com/resources/RaspberryPi/Run_Program_on_startup.pdf

The ***autoexec.sh*** file must be in the */home/pi* folder with the two lines shown below and must be executable. (chmod +x)

```
pi@raspberrypi:~ $ cat autoexec.sh
```

```
cd kiosk
```

Koisk_flip.py

sudo python kiosk_flip26.py

Any changes should be developed and tested using the x version of the file. Ie ***kiosk_flipx26.py***. See below.

The ***kiosk_flip.py*** file is the same as the ***kiosk_flip26.py***.

The files with x in them are not the FULLSCREEN versions.

For example the file ***kiosk_flipx26.py*** is identical to the

kiosk_flip26.py file except that the line:

DISPLAYSURF = pygame.display.set_mode((960, 720), pygame.FULLSCREEN)

Is commented

#DISPLAYSURF = pygame.display.set_mode((960, 720), pygame.FULLSCREEN)

Koisk_flip.py

Appendix 1 Listing the *kiosk_flip.py* file.

```
##### kiosk_flip.py

from sys import argv #used to get file name
import os
from time import *
import time
import threading
from datetime import datetime

import RPi.GPIO as GPIO
GPIO.setmode(GPIO.BCM)

GPIO.setup(18,GPIO.IN, pull_up_down=GPIO.PUD_UP)

ver ='26'

script = argv # This is the file name

print ('Version: '+ver)

import pygame
pygame.init()

screen = pygame.display.set_mode((0,800)) # Sets the screen size
screen.fill((0,0,0)) #Fills the bbackground with black
pygame.display.set_caption(str(script)+' '+ ver) # Title of window
spanish = pygame.image.load('ResourceSpanish.png').convert()
english = pygame.image.load('ResourceEnglish.png').convert()
pygame.display.set_mode((960,720),0,32) ## sets full screen
DISPLAYSURF = pygame.display.set_mode((960, 720), pygame.FULLSCREEN)
pygame.mouse.set_visible(False)
```

Koisk_flip.py

```
def show_display(message):

    font = pygame.font.Font(None,30)
    text = font.render(message, 1,OLIVE)
    screen.blit(text,(900,850)) ### Hor.Vert

disp_mode= spanish

if disp_mode == spanish:
    message = 'Click on the mounse for English'
else:
    message = 'Click en el mouse para espanol'

#####

try:
    while True:
        screen.fill(BLACK)
        screen.blit(disp_mode,(10,10))
        if disp_mode == spanish:
            message = 'Click on the mouse for English'
        else:
            message = 'Click on the mouse for Spanish'

        show_display(message)

        pygame.display.update()
        time.sleep(ti)
        for event in pygame.event.get():
            if event.type == pygame.MOUSEBUTTONDOWN:
                print("MOUSEBUTTONDOWN event occured")
                if disp_mode ==english:
                    disp_mode = spanish
                else:
```

Koisk_flip.py

```
disp_mode = english
```

```
print('disp_mode = ',disp_mode)  
show_display(message)
```

```
#####
```

```
if GPIO.input(18) == 0:  
    print ('Button Pressed')
```

```
    if disp_mode ==english:  
        disp_mode = spanish  
    else:  
        disp_mode = english
```

```
    show_display(message)
```

```
    time.sleep(.1)
```

```
except (KeyboardInterrupt, SystemExit):  
    print('The program was manually stopped')  
    quit()
```